

**DISEÑO DE UN CONTROL POR REDES NEURONALES PARA LA
OPTIMIZACIÓN DE TRAYECTORIA DE UN BRAZO ROBOT TIPO SCARA**

ANDRÉS FELIPE MINA CORRALES

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS BÁSICAS
PROGRAMA DE INGENIERÍA MECATRÓNICA
SANTIAGO DE CALI
2008**

**DISEÑO DE UN CONTROL POR REDES NEURONALES PARA LA
OPTIMIZACIÓN DE TRAYECTORIA DE UN BRAZO ROBOT TIPO SCARA**

ANDRÉS FELIPE MINA CORRALES

Trabajo de grado para optar al título de Ingeniero Mecátronico

Director
DAVID F. RAMIREZ M. M.Sc
Departamento de Física.

**UNIVERSIDAD AUTÓNOMA DE OCCIDENTE
FACULTAD DE INGENIERÍA
DEPARTAMENTO DE CIENCIAS BÁSICAS
PROGRAMA DE INGENIERÍA MECATRÓNICA
SANTIAGO DE CALI
2008**

Nota de aceptación:

Aprobado por el Comité de Grado en cumplimiento de los requisitos exigidos por la Universidad Autónoma de Occidente para optar al título de Ingeniero Mecánico

Ing. JIMMY TOMBE

JURADO

Ing. JUAN CARLOS MENA

JURADO

Santiago de Cali, 02 de Febrero de 2009

Este trabajo es dedicado a mi madre ANA CILIA y mi padre LUIS HERNANDO, a mis hermanos, familiares y amigos que estuvieron presentes en mi vida.

AGRADECIMIENTOS

Universidad Autónoma de Occidente

Grupo de Investigación de Neurocontrol Motor

David F. Ramirez M.

Julie S. Berrio P.

Jesus A. Lopez

Iván Quintero

Oscar E. León S.

Alex Payares

CONTENIDO

	Pág.
RESUMEN	10
INTRODUCCIÓN	11
1. OBJETIVOS	12
1.1. OBJETIVO GENERAL	12
1.2. OBJETIVOS ESPECIFICOS	12
2. PLANTEAMIENTO DEL PROBLEMA	13
3. MARCO TEORICO	14
3.1. CONCEPTUALIZACIÓN	14
3.2. ALGORITMO DE ENTRENAMIENTO	17
3.3. ESTRATEGIAS DE CONTROL	18
3.4. MECÁNICA DEL SISTEMA: TEORÍAS DE FALLA	19
4. MARCO TEORICO	21
4.1. ENTRENAMIENTO DE LA RED: ENTRADAS Y SALIDAS.	21
4.2. MAPAS AUTOORGANIZADOS	24
4.3. GENERACION DE LA TRAYECTORIA	27
5. CONSTRUCCIÓN MECÁNICA	30
5.1. DISEÑO MECANICO	30

5.2. MATERIALES	33
5.3. SENSORES	33
5.4. ANÁLISIS DE ESFUERZO Y DEFORMACIÓN	33
6. SIMULACIONES DE LA RED	36
7. RESULTADOS Y CONCLUSIONES	39
BIBLIOGRAFÍA	40

LISTA DE ESQUEMAS

	Pág.
Figura 1. Simulación de Obstáculos	15
Figura 2. Partes del brazo manipulador	15
Figura 3. Trayectoria con excitación de neuronas cercanas	16
Figura 4. Trayectoria utilizando el criterio de vecindad	16
Figura 5. Restricción del movimiento	17
Figura 6. Esquema básico de un Sistema de Control	18
Figura 7. Función de transferencia tipo tansig	21
Figura 8. Función de transferencia tipo logsig	21
Figura 9. Función de transferencia tipo purelin	22
Figura 10. Estructura de la red	23
Figura 11. Entrenamiento de la red (entrada, salida)	24
Figura 12. Posición inicial de las neuronas	25
Figura 13. Inicio del recorrido de las neuronas	25
Figura 14. Recorrido de neuronas	26
Figura 15. Ubicación final de las neuronas	26
Figura 16. Trayectoria calculada	27
Figura 17. Diagrama de flujo	28
Figura 18. Vecindad.	28
Figura 19. Trayectoria optima.	29
Figura 20. Diseño mecánico	30

Figura 21. Diseño explosionado	31
Figura 22. Movimiento del antebrazo	32
Figura 23. Transmisión de movimiento del brazo.	32
Figura 24. Robot Scara convencional	33
Figura 25. Restricciones	34
Figura 26. Carga aplicada	34
Figura 27. Factor de seguridad	35
Figura 28. Simulación de la red n1	36
Figura 29. Simulación de la red n2	36
Figura 30. Simulación de la red n3	37
Figura 31. Simulación de la red n4	37
Figura 32. Simulación de la red n5	38
Figura 33. Simulación de la red n6	38

RESUMEN

En la realización de este proyecto se busca la optimización de una trayectoria recorrida por un brazo artificial de dos articulaciones manteniendo las restricciones humanas en un espacio de dos dimensiones XY.

Básicamente el proyecto se presenta en tres partes donde se realizan las diferentes formas para el control de posición del brazo manipulador:

En primera instancia se muestra un entrenamiento de una red neuronal de dos capas ocultas y una de salida donde se presentan los patrones de entrada (XY numero de posiciones finitos) y salidas (los ángulos de las dos articulaciones). En segundo lugar se realiza una autoorganización de neuronas en una región de trabajo utilizando la regla de aprendizaje de kohonen y se calcula la trayectoria utilizando cinemática inversa, luego se presenta una excitación de las neuronas vecinas a la trayectoria calculada e inhibiciones a los nodos que no presenten cercanía. Por ultimo se presenta una red neuronal que realiza una autoorganización de neuronas en su espacio de trabajo y luego genera la trayectoria del brazo mecánico.

Se realiza un diseño mecánico del brazo donde se analizan los esfuerzos y las deformaciones presentes en la zona de trabajo. También se muestra una simulación del movimiento de las dos articulaciones.

INTRODUCCIÓN

En la actualidad se emplean varias estrategias de control para el posicionamiento de un robot, como se aprecia en el Proportional Integral Derivative (PID). También se emplea el control por par conmutado, el control predictivo funcional y el control por redes neuronales artificiales (RNA).

En las últimas décadas, las RNA se han convertido en un campo interdisciplinario de gran desarrollo, con múltiples aplicaciones actuales y potenciales. Hoy en día son muy usadas en el área militar, para el reconocimiento de objetivos, control de trayectorias de misiles, robots buscaminas; en el área aeroespacial para el control de robots, estudio de cuerpos estelares; en el área de ingenierías para el diseño de sistemas de control adaptativos, visión artificial, robótica móvil, procesamiento paralelo de datos, robots manipuladores, etc.

En este trabajo nos centramos en el control de un brazo robótico tipo Scara (Selective appliance arm robot for assembly) por medio de redes neuronales artificiales, ya que por medio de la emulación del control motor de los primates, podemos tomar un modelo e implementarlo en el control de un robot manipulador.

1. OBJETIVOS

1.1. OBJETIVO GENERAL

- Diseñar una red neuronal artificial para el control óptimo de la trayectoria de una brazo robótico tipo Scara para el alcance de una posición predeterminada.

1.2. OBJETIVOS ESPECIFICOS

- Diseñar un brazo mecánico de dos articulaciones utilizando el software de Solid Edge.
- Diseñar un controlador por redes neuronales artificiales utilizando el software de MatLab.
- Diseñar una animación del brazo mecánico utilizando el software de Solid Edge.
- Escribir un informe final del trabajo realizado en formato IFAC.

2. PLANTEAMIENTO DEL PROBLEMA

La realización de un control de posición de un brazo manipulador posee una gran complejidad, debido a la no-linealidad y a la optimización de trayectorias para un funcionamiento eficiente, y también las características mecánicas de los sistemas que restringen la realización óptima del robot.

Existen una serie de dificultades en el control de posición de un brazo manipulador que conllevan a un funcionamiento más complejo de la estrategia de control utilizada. Cuando se va a realizar un movimiento desde un punto inicial a un punto final, se calcula una trayectoria óptima teniendo en cuenta los posibles obstáculos presentes en la zona de trabajo. Dicha trayectoria calculada puede ser una línea continua de puntos que hace difícil la tarea del robot manipulador, provocando así un margen de error no deseado. En muchos casos los brazos manipuladores se flectan debido a la carga presente, esto produce una mala precisión del sistema.

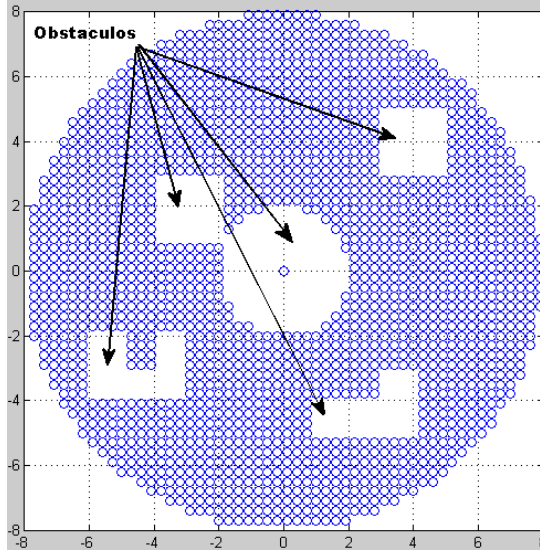
3. MARCO TEORICO

3.1. CONCEPTUALIZACIÓN

En el recorrido realizado por el brazo mecánico de dos articulaciones (brazo y antebrazo) desde un punto inicial hasta un punto final conformando una trayectoria óptima en una zona de trabajo presente de obstáculos, es necesario tener en cuenta los siguientes conceptos:

- **Identificación de la región de trabajo:** El robot manipulador actúa sobre una región de trabajo que se encuentra descrita por un código poblacional neuronal, con nodos o unidades de cómputo ubicadas en posiciones XY que se corresponden con los pesos sinápticos. Teniendo en cuenta las restricciones mecánicas del brazo, el área de trabajo esta presente en forma de anillo donde su radio máximo es la distancia de las dos extremidades (distancia entre hombro, codo y mano).
- **Simulación de obstáculos:** En la región de trabajo del robot, se presenta una serie de obstáculos que son representados por un grupo de neuronas inhibidas. Dichas neuronas no presentaran activación para la generación de la trayectoria.

Figura 1. Simulación de Obstáculos



- **Generación de la trayectoria:** Para la generación de la trayectoria se optó por dos alternativas; utilizando un mapa autoorganizado se genera una región de trabajo y luego se calcula la trayectoria utilizando cinemática inversa. Se presenta una excitación de las neuronas de la zona de trabajo que se encuentren cercanas a la trayectoria calculada.

Figura 2. Partes del brazo manipulador

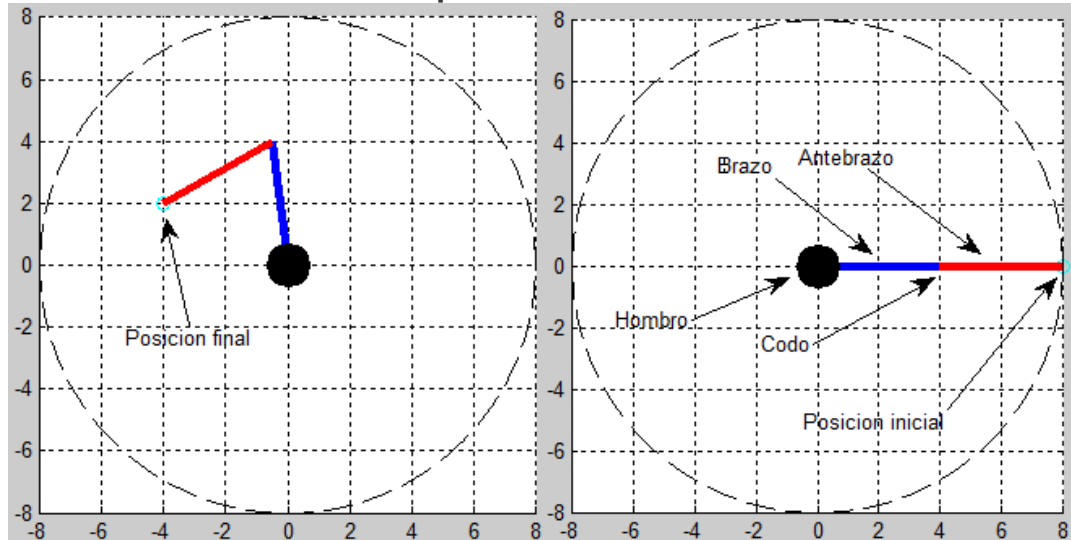


Figura 3. Trayectoria con excitación de neuronas cercanas

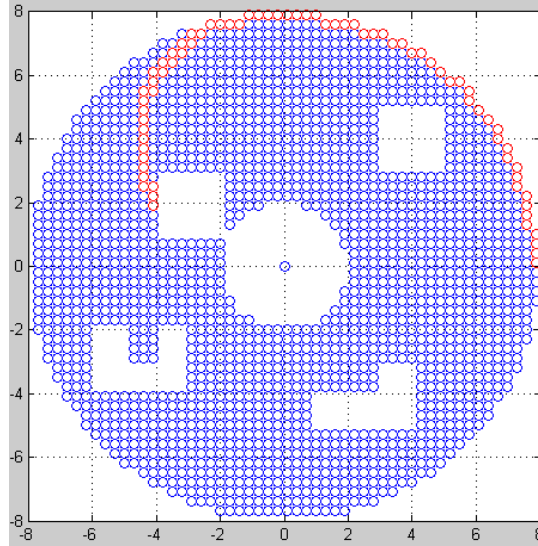
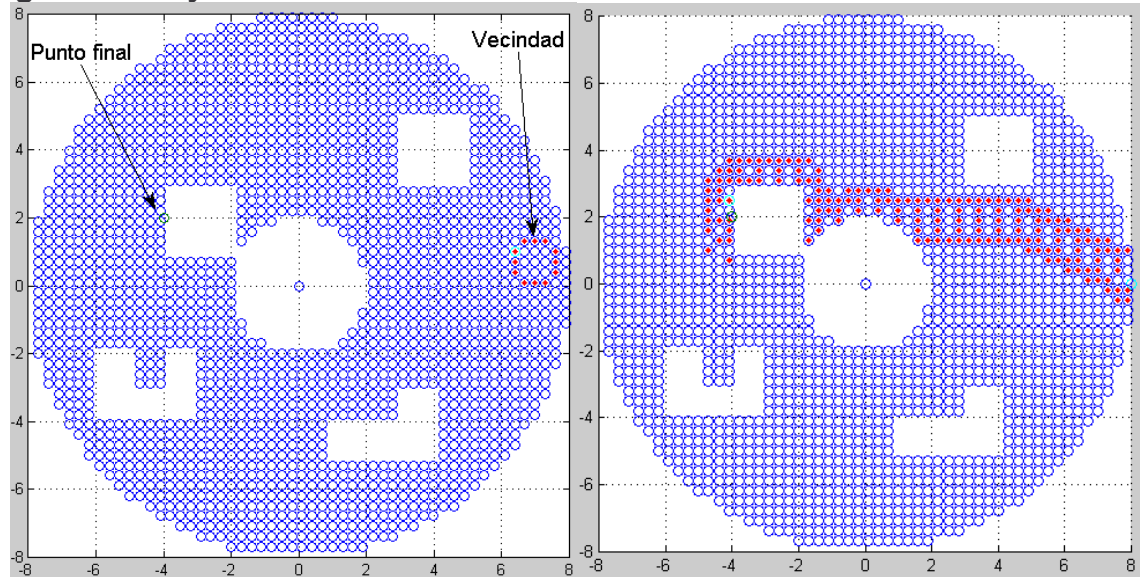
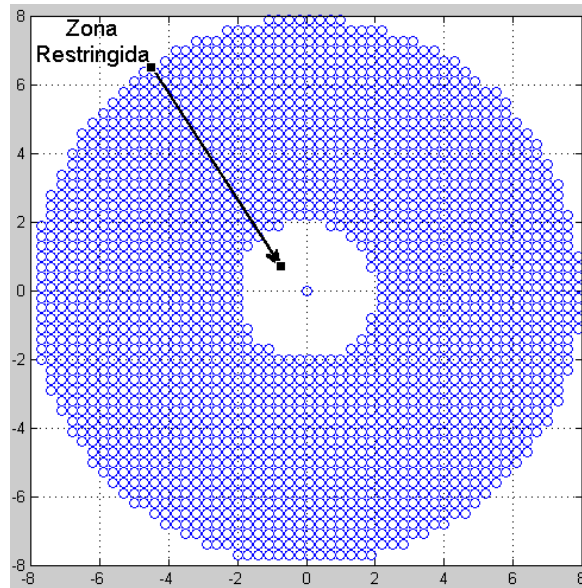


Figura 4. Trayectoria utilizando el criterio de vecindad



- **Restricción de movimiento:** La zona de trabajo es presentada en forma de anillo donde el círculo inferior representa un área restringida. Esto es debido a la dinámica del sistema presente en el brazo.

Figura 5. Restricción del movimiento



- **Movimiento de las articulaciones:** El movimiento efectuado por el brazo es realizado de acuerdo a la posición de la mano. Cuando se obtiene un punto XY, se utiliza cinemática inversa para calcular los ángulos del brazo y antebrazo.

3.2. ALGORITMO DE ENTRENAMIENTO

Los algoritmos de entrenamiento son utilizados en distintas estrategias de control y en aquellas instrumentadas por redes neuronales artificiales. El algoritmo de Levenberg-Marquardt (LM) utilizando el método de backpropagation es utilizado en este trabajo, en el entrenamiento de dos redes neuronales para la generación de los ángulos de las articulaciones. Este algoritmo realiza un ajuste de los pesos sinápticos utilizando una regla de minimización del error (LM). Otros algoritmos de entrenamiento son utilizados en las redes neuronales como es el Gradiente Descendente (GD) también conocido como el método de retropropagación del error que requiere un importante número de cálculos para lograr el ajuste de los pesos de la red. En la implementación del algoritmo del GD, se toma una amplitud de pasos que viene dada por una tasa de aprendizaje. A mayor tasa de aprendizaje, mayor es la modificación de los pesos en cada iteración con lo que el gradiente descendente es más rápido (Hileras y Martínez, 1995).

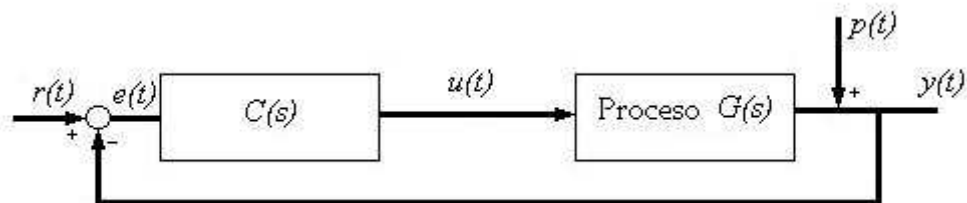
3.3. ESTRATEGIAS DE CONTROL

- De acuerdo con *PID Proportional Integral Derivative*, (Tacconi, Mantz, Solsona y Pulesto 2005), “los robots industriales utilizan normalmente el control clásico tipo PID. Pero los brazos manipuladores, los cuales se componen de varias articulaciones unidas entre sí, poseen una dinámica altamente no lineal con un fuerte acoplamiento entre sus respectivas articulaciones. Esto complica la tarea de un simple controlador PID así como su sintonización, sobre todo, con consignas a altas velocidades o aceleraciones”.

De los mismos autores: “Las estrategias de control PID se incorporan en el ambiente industrial en el primer cuarto de este siglo, con un esquema de ajuste puramente empírico. En la actualidad, y pese al sorprendente desarrollo de la teoría de control y del soporte tecnológico necesario para su implementación, el controlador de estructura PID se emplea casi con exclusividad en el ambiente industrial de todo el mundo, en particular para controlar procesos térmicos y químicos”.

“El controlador PID surge como consecuencia de la combinación de tres acciones básicas de control -acción proporcional, integral y derivativa- cuyos efectos ya eran conocidos a principios de siglo. A los fines de recordar los efectos de estas tres acciones considérese el esquema de control indicado en la figura 6”.

Figura 6. Esquema básico de un Sistema de Control



“Donde $G(s)$ representa el proceso cuya variable de salida $y(t)$ se desea controlar y $C(s)$ al controlador. Asimismo $r(t)$ es la señal de referencia, o “set-point”, y $p(t)$ representa posibles perturbaciones”.

- *Lógica difusa*: Según el autor Ricardo Gutiérrez de la universidad Nacional de Colombia: “La lógica difusa tiene como ventaja principal su simplicidad para trabajar problemas complejos como lo son los controladores de sistemas no

lineales. Por medio de reglas semánticas establecidas por un experto, se puede compensar los parámetros desconocidos de un modelo, manejar las no linealidades y combinar esto con la robustez de técnicas convencionales como el PI y el PID”.

“La lógica difusa se puede utilizar para compensar parámetros desconocidos en sistemas altamente no lineales, Otra aplicación importante de la lógica difusa es ser utilizada en combinación con sistemas de control convencional como PI”.

“Debido a las propiedades de la lógica difusa el modelo del sistema no siempre es requerido, sino que las reglas construidas con ayuda de un experto son suficientes, es decir, con este tipo de controlador es más importante que conocer el comportamiento del sistemas, es conocer el comportamiento que quiero de él, esto es válido para distintos tipos de plantas incluidos los robots articulados”.

- *Control con Redes Neuronales:* Las redes neuronales han evolucionado en los últimos años debido a su gran capacidad de adaptación y a la solución de problemas de alta complejidad, como son el reconocimiento de patrones, planificación de trayectorias, optimización de trayectorias y otro tipo de tareas que necesiten un alto nivel de procesamiento. Gracias a la estructura distribuida de las redes neuronales se puede realizar un procesamiento paralelo de datos que permite realizar operaciones simultáneamente.

Para la generación de la trayectoria del robot manipulador se utilizó una estrategia de control por redes neuronales, dicha estrategia nos permitió realizar un entrenamiento utilizando el algoritmo (LM) teniendo como resultado una red neuronal, que de acuerdo a una entrada XY nos da como salida los ángulos deseados de las articulaciones.

3.4. MECÁNICA DEL SISTEMA: TEORÍAS DE FALLA

Cuando se realiza el análisis mecánico al robot manipulador se utiliza el criterio de de falla del esfuerzo cortante máximo (Tresca) propuesto por Charles Coulomb (1773) y Henri Tresca (1868), la cual se puede establecer que: “Para todo elemento fabricado del mismo material, la falta por fluencia ocurre cuando el esfuerzo cortante máximo en el material alcanza el valor del esfuerzo cortante de fluencia determinado por un ensayo uniaxial de tensión (o compresión)”.

$$\tau_{m\acute{a}x} = \tau_y = \frac{\sigma_y}{2} \quad (1)$$

“Donde σ_y es el esfuerzo de fluencia y τ_y es el esfuerzo cortante de fluencia que indica el valor crtico por usarse en el criterio de falla por esfuerzo cortante mximo”.

“La teora del esfuerzo cortante mximo extiende este resultado al caso general, que incluye un estado arbitrario de esfuerzo, estableciendo que la falla ocurre cuando el esfuerzo cortante mximo sobre cualquier plano y punto de la estructura alcanza el valor τ_y . Cuando los tres esfuerzos principales $\sigma_1 > \sigma_2 > \sigma_3$, se conocen en un punto, el esfuerzo cortante mximo est dado por”:

$$\tau_y = \frac{\sigma_1 - \sigma_3}{2} \quad (2)$$

Y la falla ocurre cuando

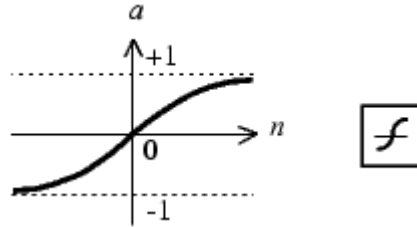
$$\frac{\sigma_1 - \sigma_3}{2} = \tau_y = \frac{\sigma_y}{2} \quad (3)$$

4. MARCO TEORICO

4.1. ENTRENAMIENTO DE LA RED: ENTRADAS Y SALIDAS.

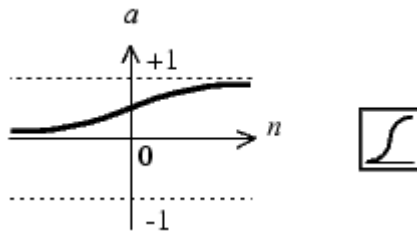
Para la generación de los ángulos de las articulaciones se utilizaron dos redes neuronales artificiales cada una conformada por dos capas ocultas y una capa de salida, (Figura10). La primera capa oculta esta conformada por 20 neuronas con función de transferencia tipo *tansig* (4), la segunda capa oculta presenta 10 neuronas con función de transferencia tipo *logsig* (5). La capa de salida solo presenta una sola neurona con función de transferencia tipo *purelin* (6).

Figura 7. Función de transferencia tipo tansig



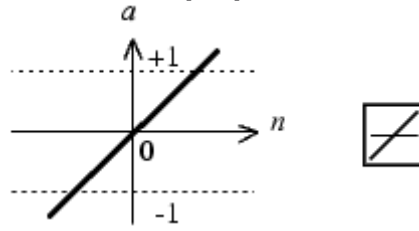
$$a = \frac{2}{1 + \exp^{(-2n)}} - 1 \quad (4)$$

Figura 8. Función de transferencia tipo logsig



$$a = \frac{1}{1 + \exp^{(-n)}} \quad (5)$$

Figura 9. Función de transferencia tipo purelin



$$a = n \quad (6)$$

Las redes neuronales fueron entrenadas y simuladas en MatLab y el Neural Network Toolbox: Se pudo realizar el entrenamiento utilizando el algoritmo trainlm (Levenberg-Marquardt). Los patrones de entrenamiento para la red fueron los posibles ángulos del brazo y antebrazo como salida, se calcularon las posiciones correspondientes X , ec. (7) y Y ec. (8) como entradas.

Para los ángulos del antebrazo $ang_1 = 0:0.1:2 \bullet pi$.

Para los ángulos del brazo $ang_2 = 0:0.1:2 \bullet pi$.

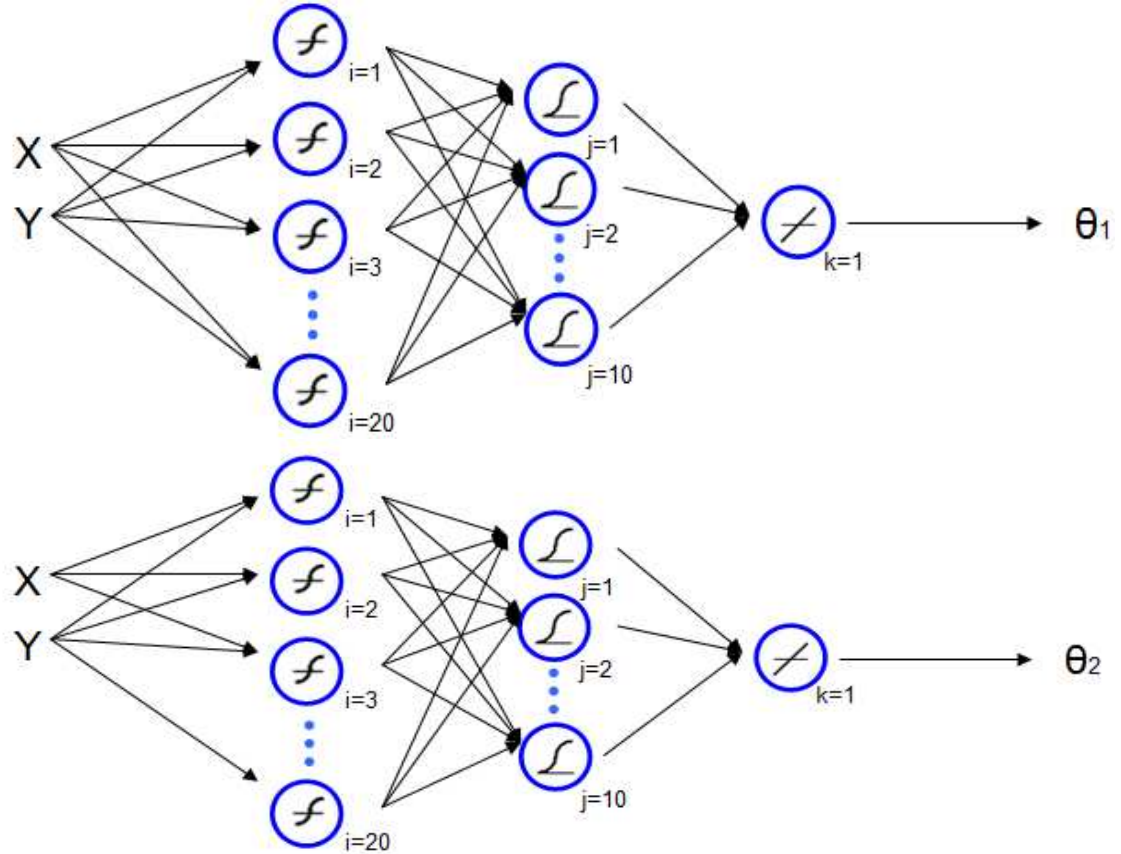
Se acomodan los ángulos $[an_1, an_2] = meshgrid(ang_1, ang_2)$.

$$x = L * \cos(an1) + L * \cos(an1 + an2) \quad (7)$$

$$y = L * \sin(an1) + L * \sin(an1 + an2) \quad (8)$$

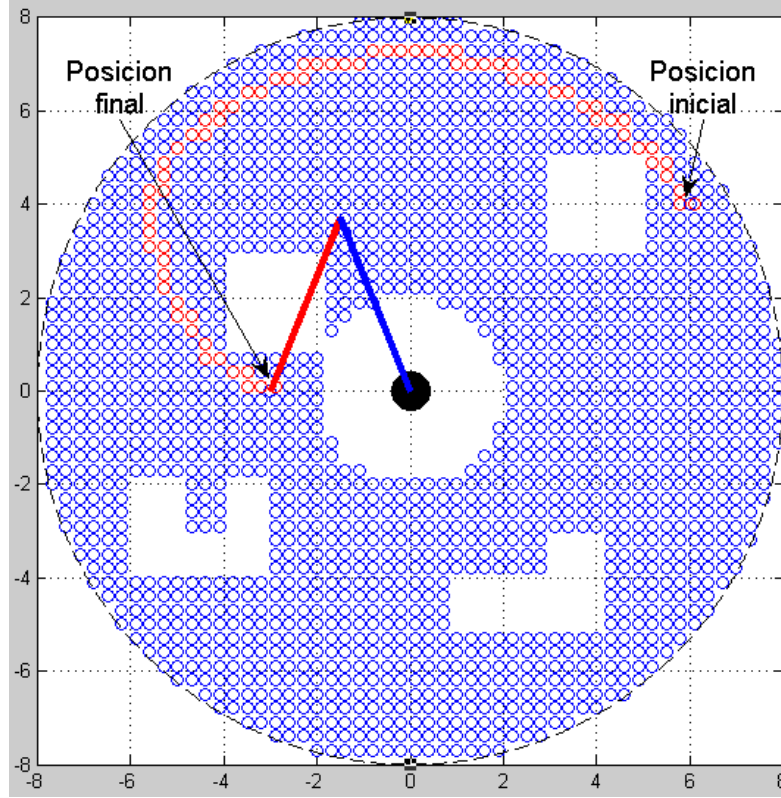
Donde L es la longitud del antebrazo e igualmente del brazo.

Figura 10. Estructura de la red



El funcionamiento general de la red es calcular inicialmente los ángulos de las articulaciones en la posición inicial y final de acuerdo a las entradas $[(X_{ini}, Y_{ini})(X_{fin}, Y_{fin})]$. Cuando se obtiene los ángulos iniciales y finales se genera el movimiento de las articulaciones. Se inicia con el movimiento de antebrazo y luego el brazo.

Figura 11. Entrenamiento de la red (entrada, salida)



4.2. MAPAS AUTOORGANIZADOS

Otra forma que se utilizó para la generación de la trayectoria fue con mapas autoorganizados. Utilizando la regla de aprendizaje de kohonen (9) se logra realizar una ubicación de representación de neuronas en un espacio de trabajo en forma de anillo.

$$\begin{aligned} W_{ux}(i) &= x(i) + \frac{1}{i}(e(1,i) - x(i)) \\ W_{uy}(i) &= y(i) + \frac{1}{i}(e(2,i) - y(i)) \end{aligned} \quad (9)$$

Donde, el término $\frac{1}{i}$ es un parámetro de ganancia o coeficiente de aprendizaje, con un valor entre 0 y 1 que decrece con el número de iteraciones del proceso de entrenamiento y $e(i)$ es el vector de entrenamiento.

Inicialmente se encuentran ubicadas una representación de neuronas en la posición (0,0) y luego se distribuyen en el espacio de trabajo. Si la ubicación final de una neurona está presente en una zona donde hay un obstáculo, dicha neurona no presenta movimiento manteniendo su posición inicial.

Figura 12. Posición inicial de las neuronas

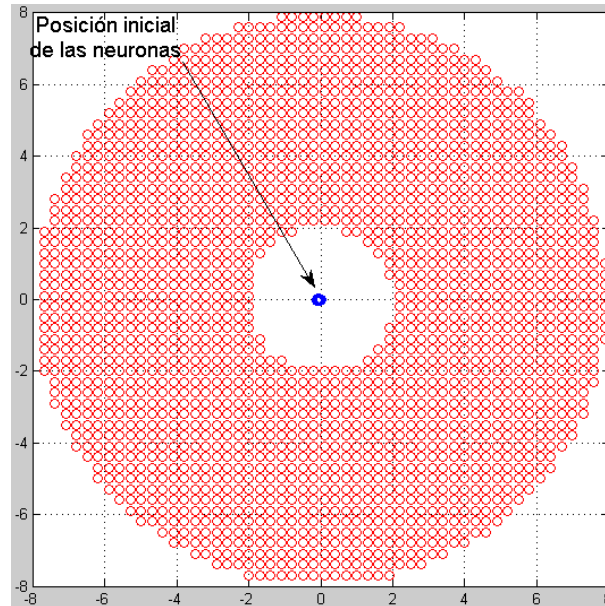


Figura 13. Inicio del recorrido de las neuronas

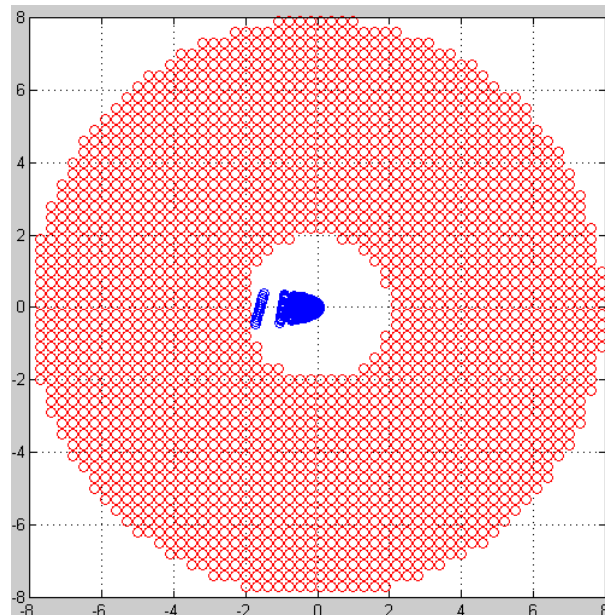


Figura 14. Recorrido de neuronas

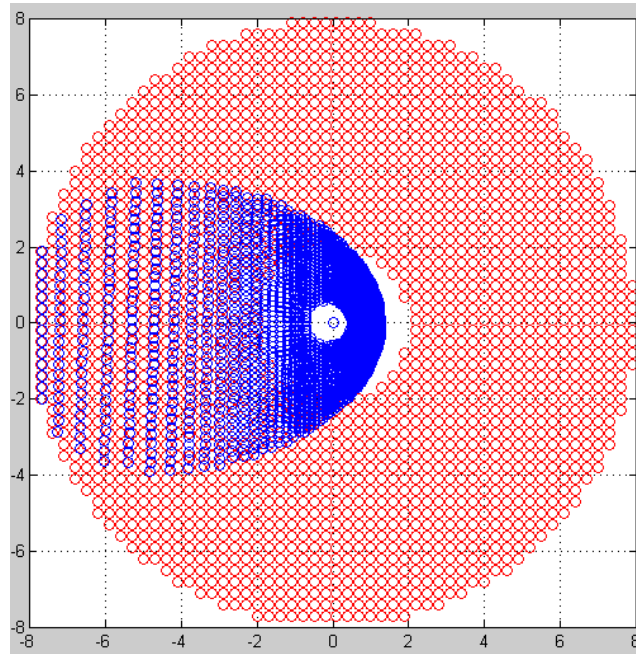
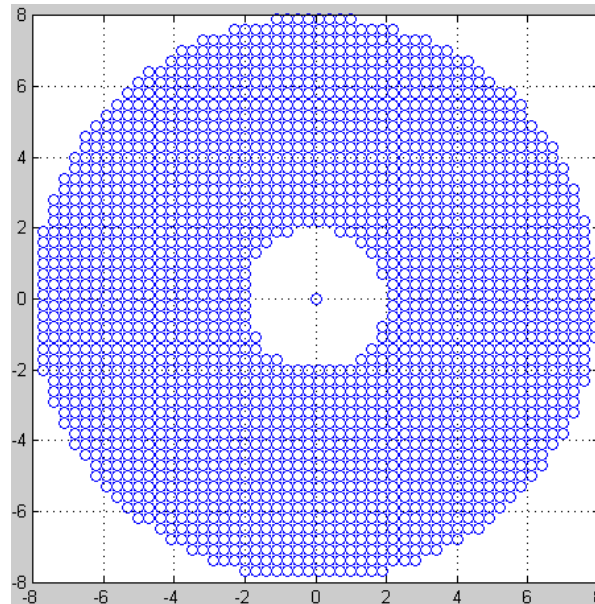
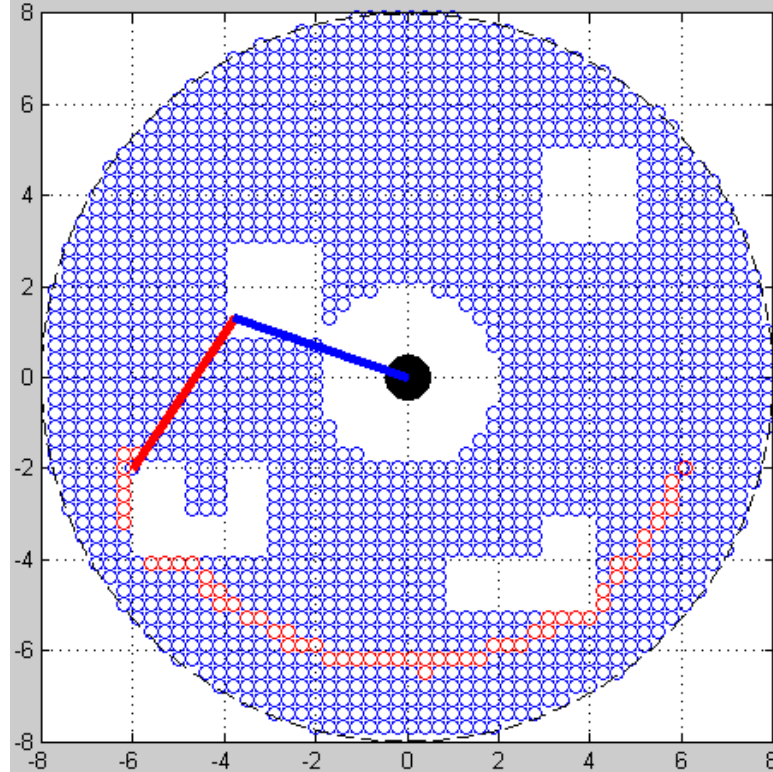


Figura 15. Ubicación final de las neuronas



Cuando se tiene ubicadas la representación de neuronas en el espacio XY, se calcula la trayectoria del brazo. Dicha trayectoria es calcula utilizando cinemática inversa. Luego se realiza una excitación de neuronas que presentan vecindad con la trayectoria calculada e inhibición al resto.

Figura 16. Trayectoria calculada



4.3. GENERACION DE LA TRAYECTORIA

Por ultimo se realiza un cálculo de la trayectoria del robot manipulador tipo Scara por medio de una red neuronal utilizando el criterio de vecindad. Conociendo la posición inicial se realiza una excitación de las neuronas vecinas, cada una de ellas calcula la distancia que hay con la meta, la neurona vecina que tenga la distancia mas corta será la ganadora, la nueva neurona calcula de nuevo la vecindad, este proceso se realiza hasta que la distancia calculada sea la minima. La ubicación de la representación de neuronas en el espacio de trabajo fueron generadas con el misma ecuación (9) expuesta en la unidad anterior.

Figura 17. Diagrama de flujo

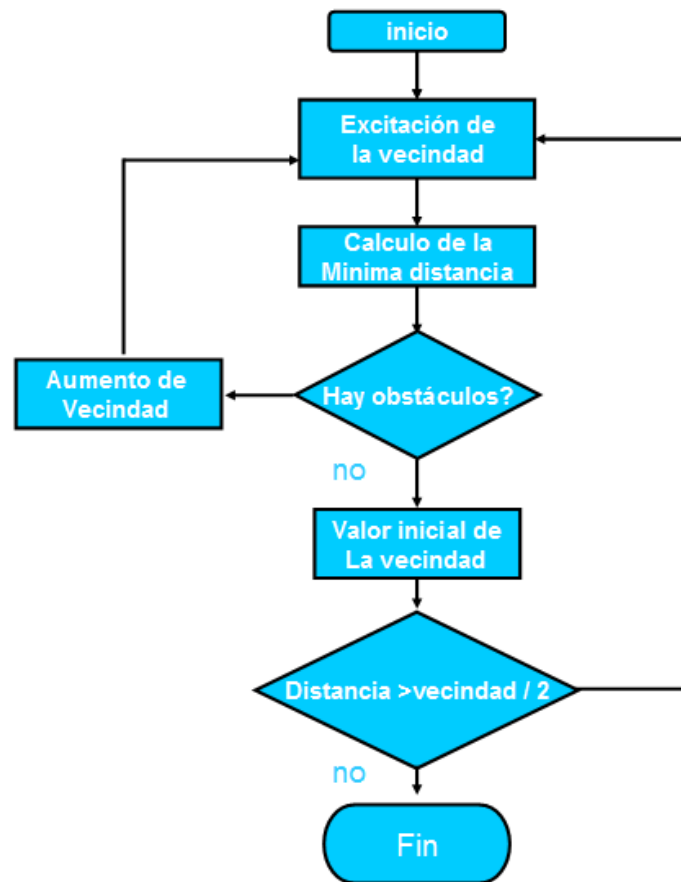


Figura 18. Vecindad.

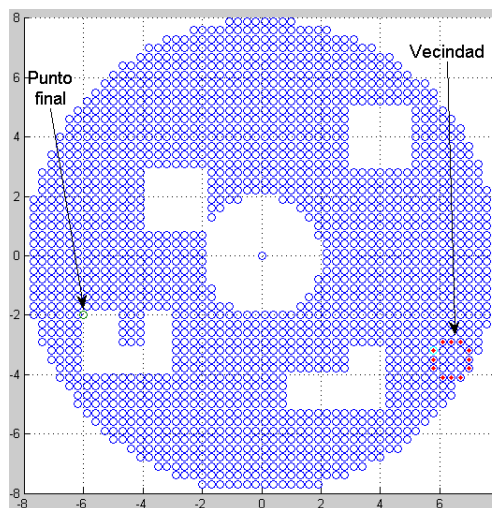
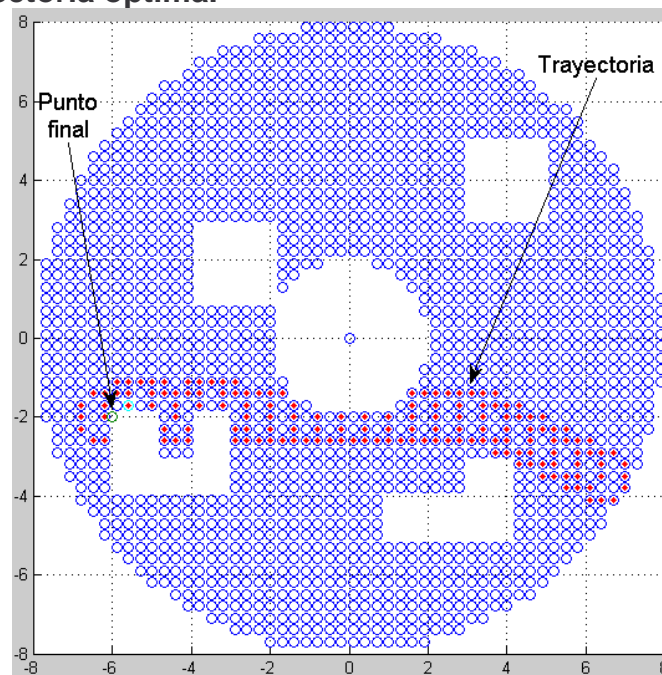


Figura 19. Trayectoria optima.



Dicho proceso en general comparado con los anteriores, realiza una mejor optimización en el cálculo de la trayectoria para la eficiencia del brazo manipulador.

5. CONSTRUCCIÓN MECÁNICA

5.1. DISEÑO MECANICO

Una vez terminado el control por redes neuronales para el posicionamiento del brazo, se realiza el diseño mecánico gracias a la ayuda del software Solid Edge V20.

Figura 20. Diseño mecánico

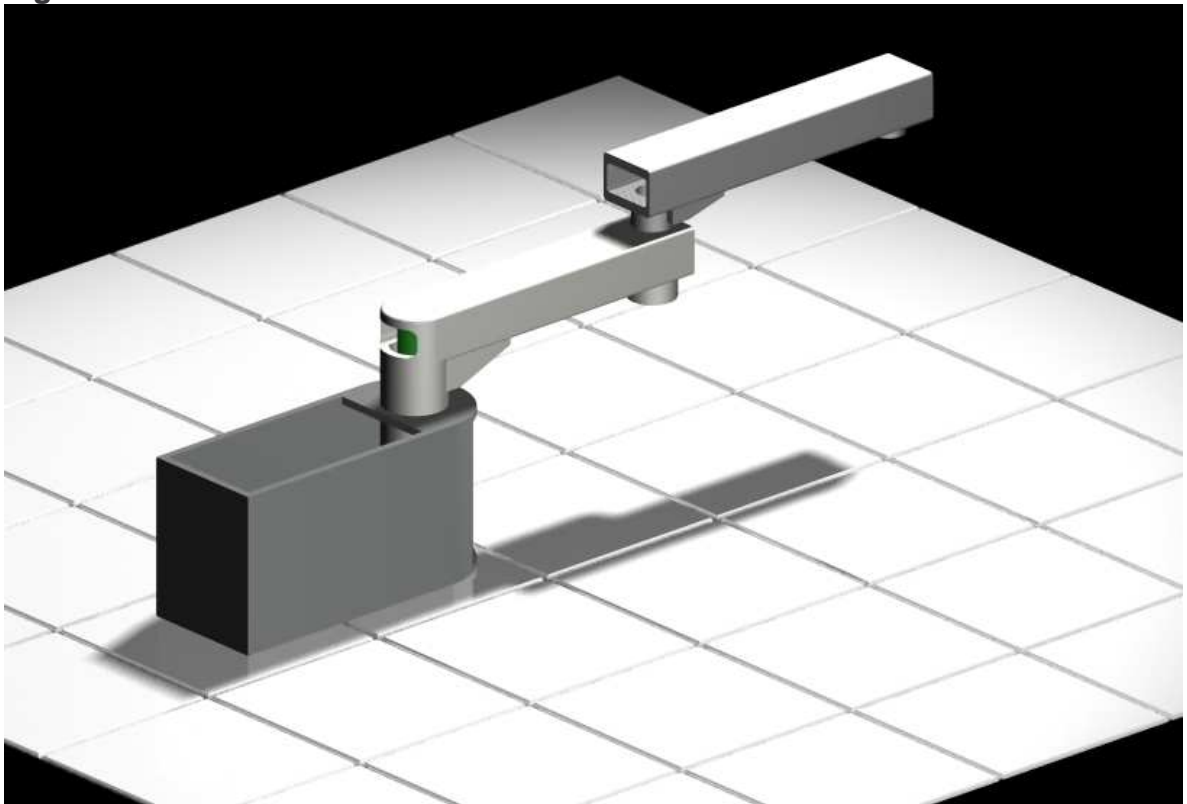
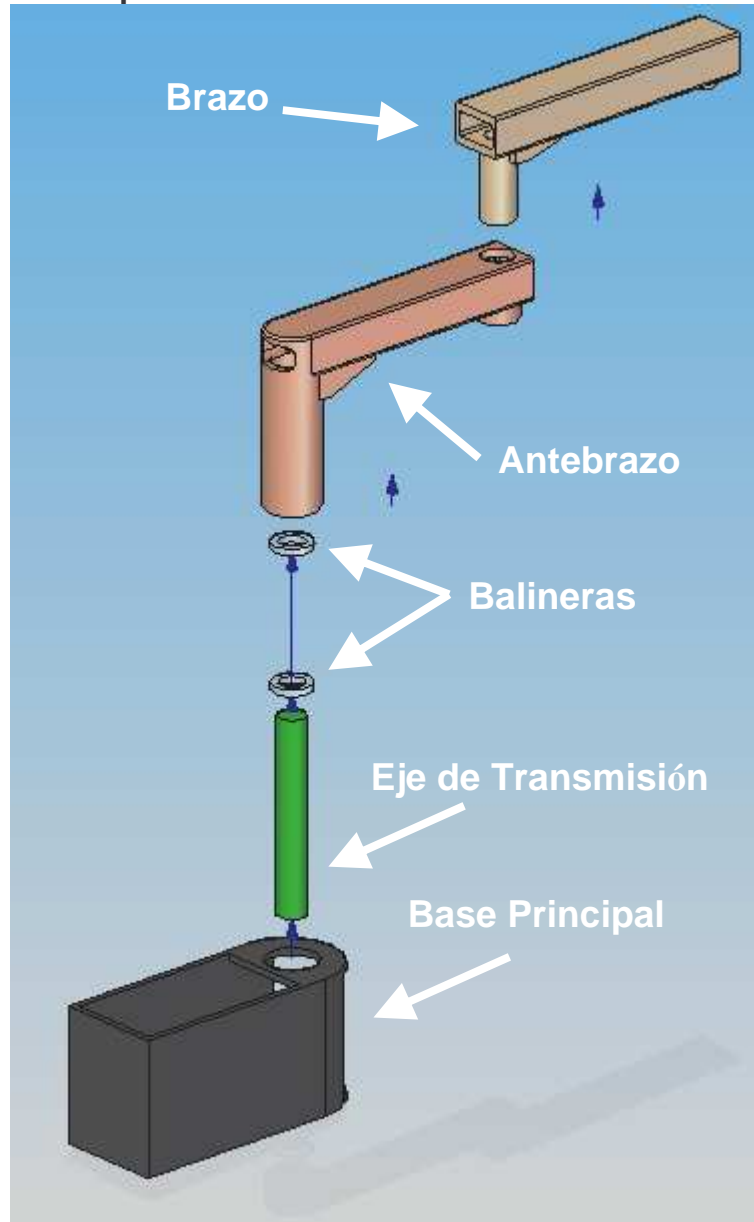
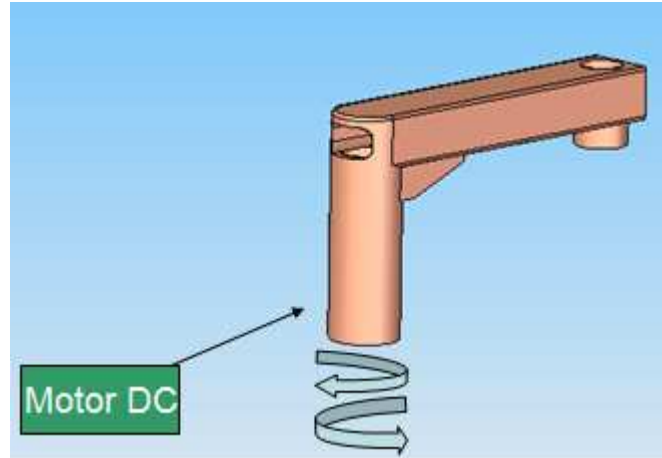


Figura 21. Diseño explosionado



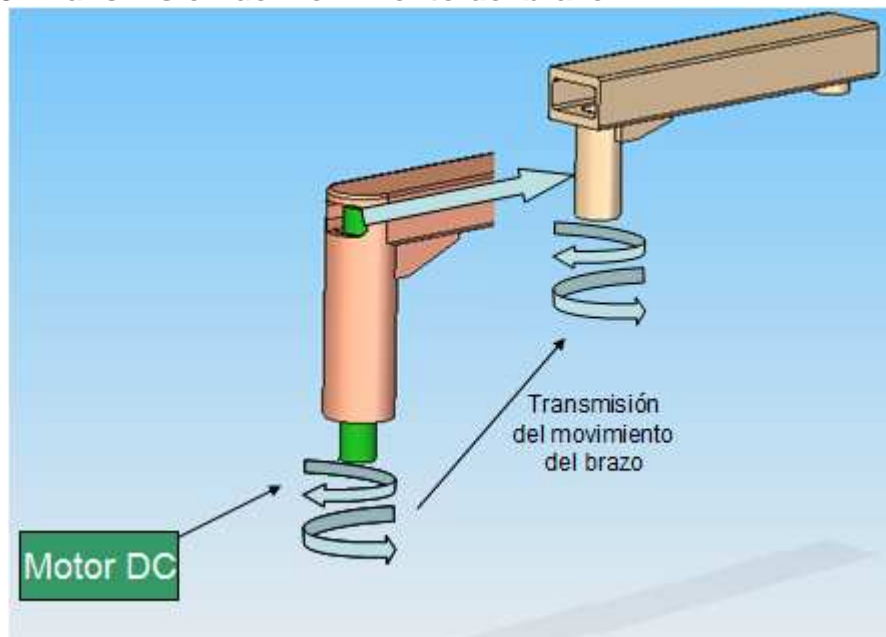
El diseño mecánico esta constituido por 5 partes ensambladas entre si. Para la mayoría de las piezas el material utilizado fue acero estructural. El brazo mecánico esta diseñado para moverse sobre un solo plano. El movimiento es efectuado por dos motores DC que se encuentran ubicados en la base principal, cada motor posee una caja reductora para el aumento del torque necesario. Uno de los motores será el encargado de mover el antebrazo Figura 22, el motor esta conectado al antebrazo por medio de una correa dentada.

Figura 22. Movimiento del antebrazo



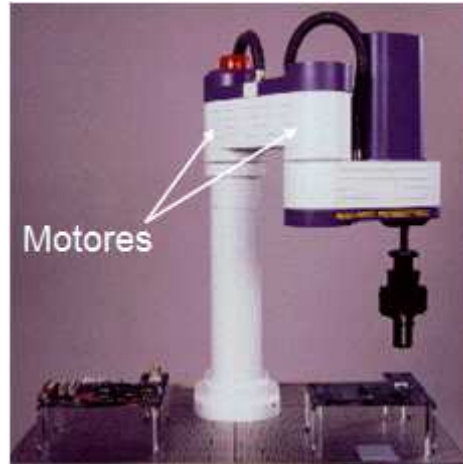
El segundo motor realiza el movimiento del brazo, para la transmisión del movimiento se utiliza una correa dentada entre el motor y la parte inferior del eje de transmisión, luego por medio de otra cadena conectada entre al eje de transmisión en la parte superior y el brazo Figura 23.

Figura 23. Transmisión de movimiento del brazo.



Este sistema de transmisión es diferente a los convencionales, en donde los motores están conectados directamente y sobre las articulaciones Figura 24.

Figura 24. Robot Scara convencional



5.2. MATERIALES

Para el diseño de brazo robot se utilizó un Steel AISI 1080 As-rolled con alto porcentaje de carbono. Una característica principal de este material es su dureza y sus propiedades mecánicas. Para el resto de las piezas se utilizó un Steel ASTM A-36, este acero estructural no es frágil, presenta buena ductilidad y sus propiedades mecánicas son óptimas en la zona de trabajo.

5.3. SENSORES

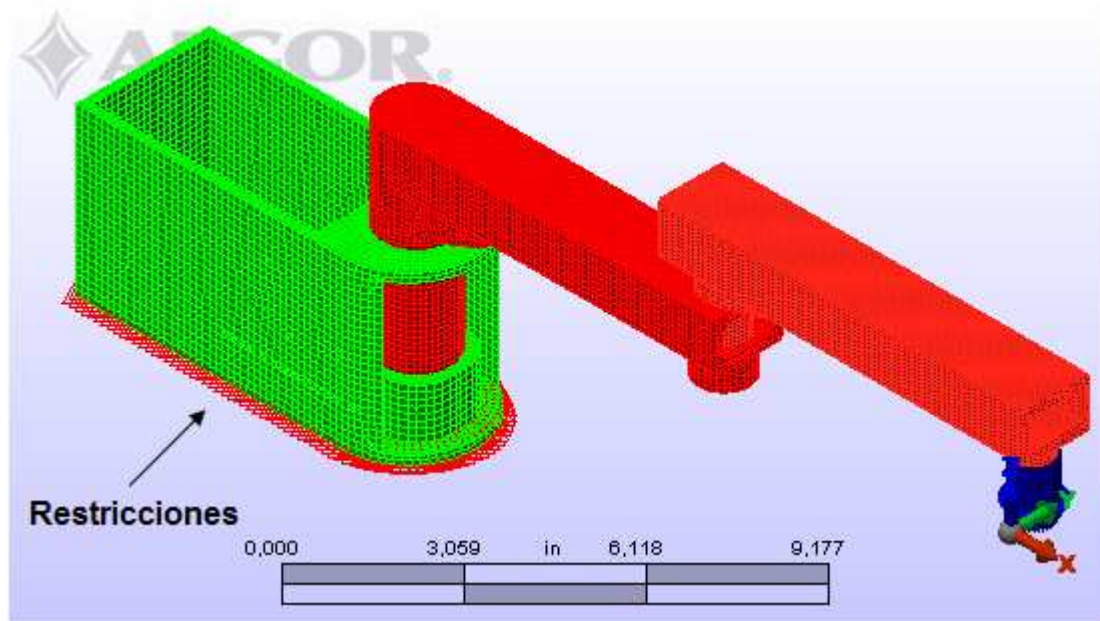
Para la lectura de las variables del robot (que en este caso son los ángulos formados entre los segmentos articulados) se utilizan potenciómetros lineales. Los potenciómetros lineales son los encargados de generar un voltaje de acuerdo a una posición angular, esto se debe a que internamente el potenciómetro realiza un divisor de tensión para la regulación del voltaje. En este trabajo los potenciómetros están ubicados en los ejes de las articulaciones.

5.4. ANÁLISIS DE ESFUERZO Y DEFORMACIÓN

Utilizando el software se Algor V22 se realizó un análisis mecánico del brazo. El robot se analizó como elemento tipo Brick y se seleccionaron los materiales adecuados.

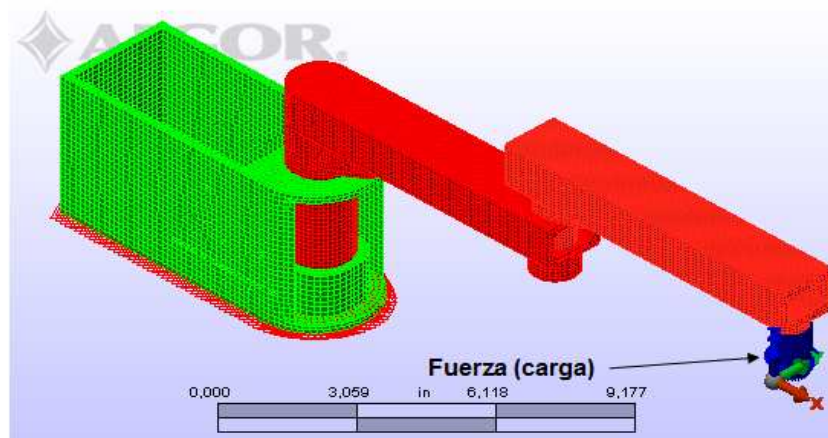
Las restricciones fueron aplicadas sobre la plataforma inferior de la base principal Figura 25.

Figura 25. Restricciones



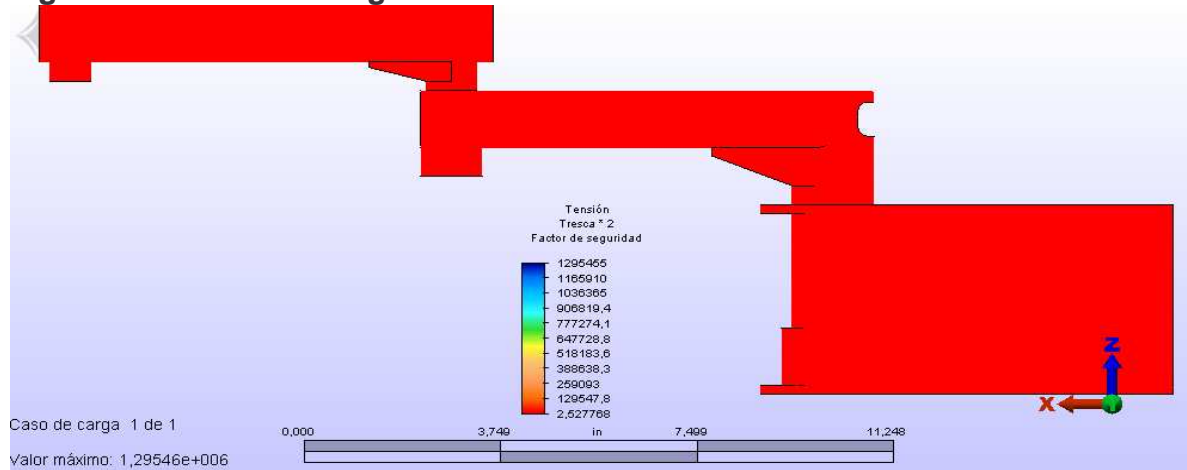
El brazo se encuentra en su peor posición debido a que se encuentra totalmente estirado. Al final del brazo (mano) se le aplico una carga de 1 lbf, esta seria la máxima carga que el brazo pueda llevar durante su recorrido Figura 26.

Figura 26. Carga aplicada



Finalmente se realizó un análisis de Tresca para obtener el factor de seguridad (FS) del brazo. Como podemos apreciar en la figura 27 el mínimo factor de seguridad fue de 2,52. Este resultado fue favorable ya que lo indicado para este diseño fue de 2. Este resultado nos dice que el brazo presentará un buen comportamiento en la zona de trabajo.

Figura 27. Factor de seguridad



6. SIMULACIONES DE LA RED

Figura 28. Simulación de la red n1

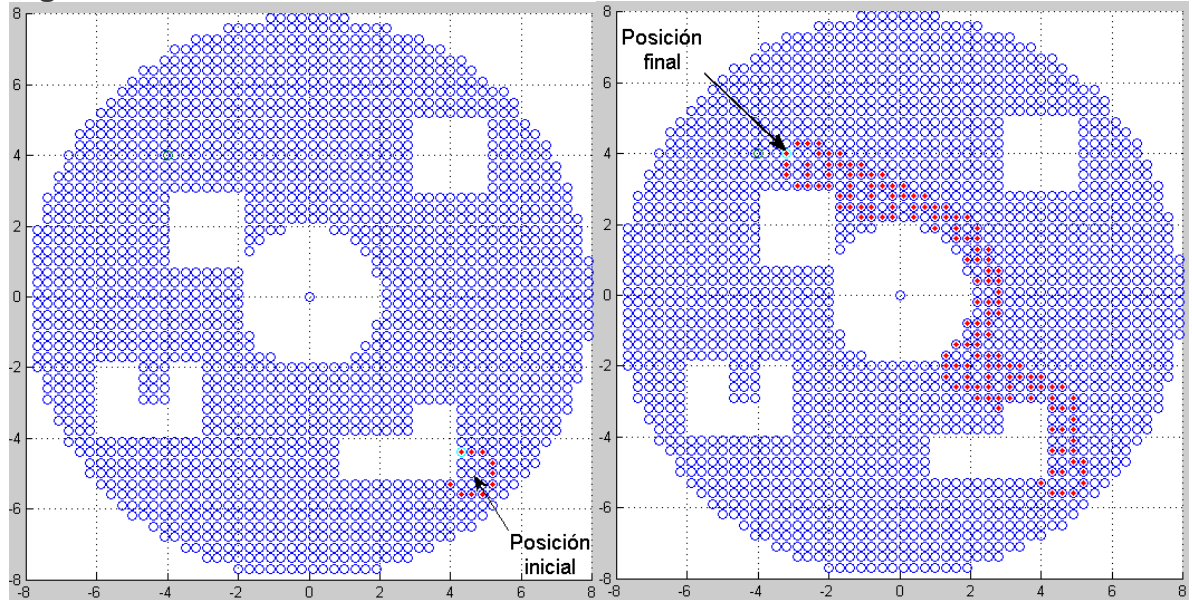


Figura 29. Simulación de la red n2

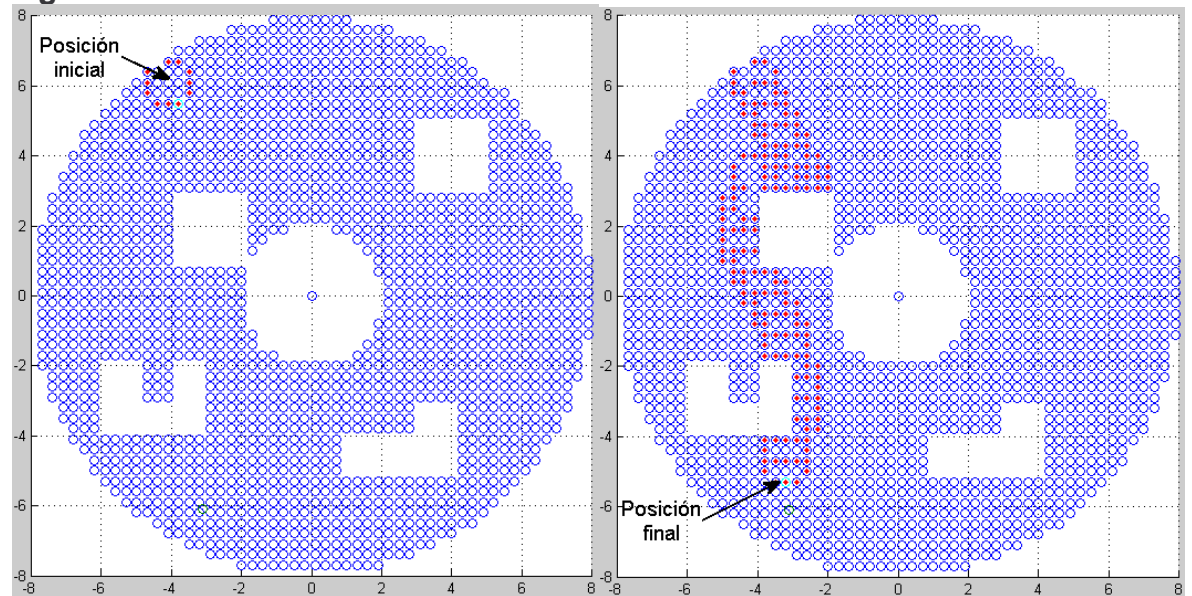


Figura 30. Simulación de la red n3

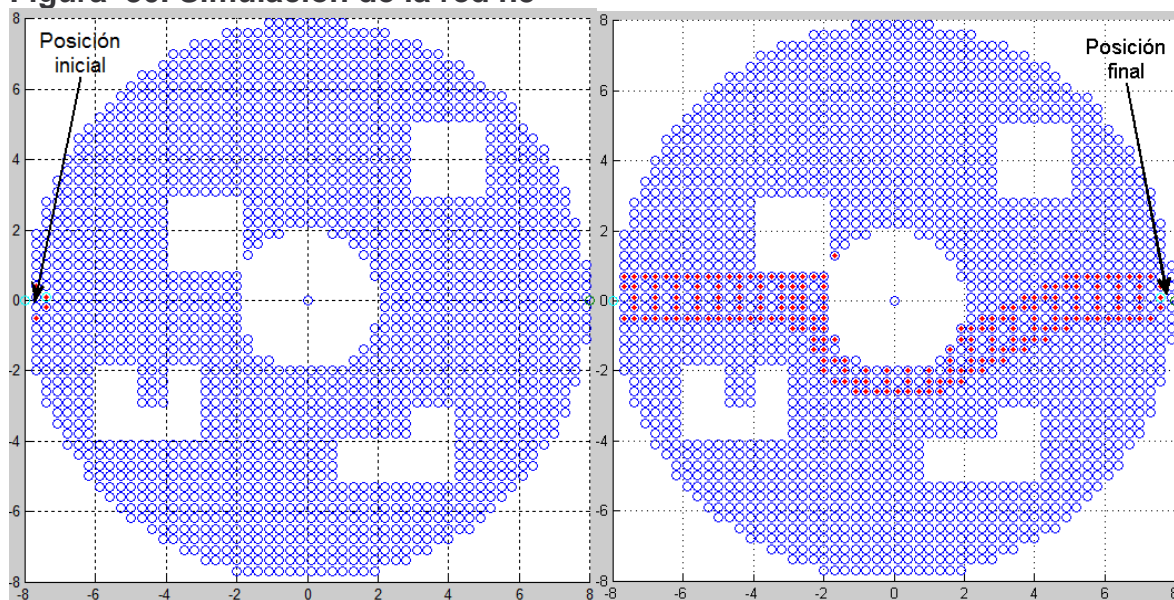


Figura 31. Simulación de la red n4

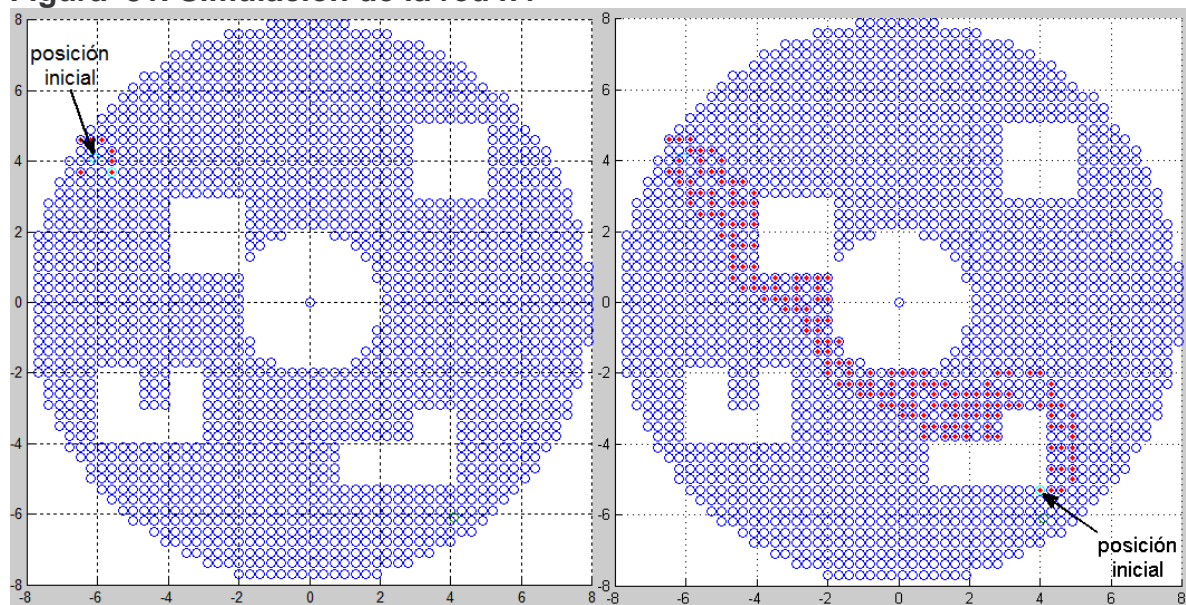


Figura 32. Simulación de la red n5

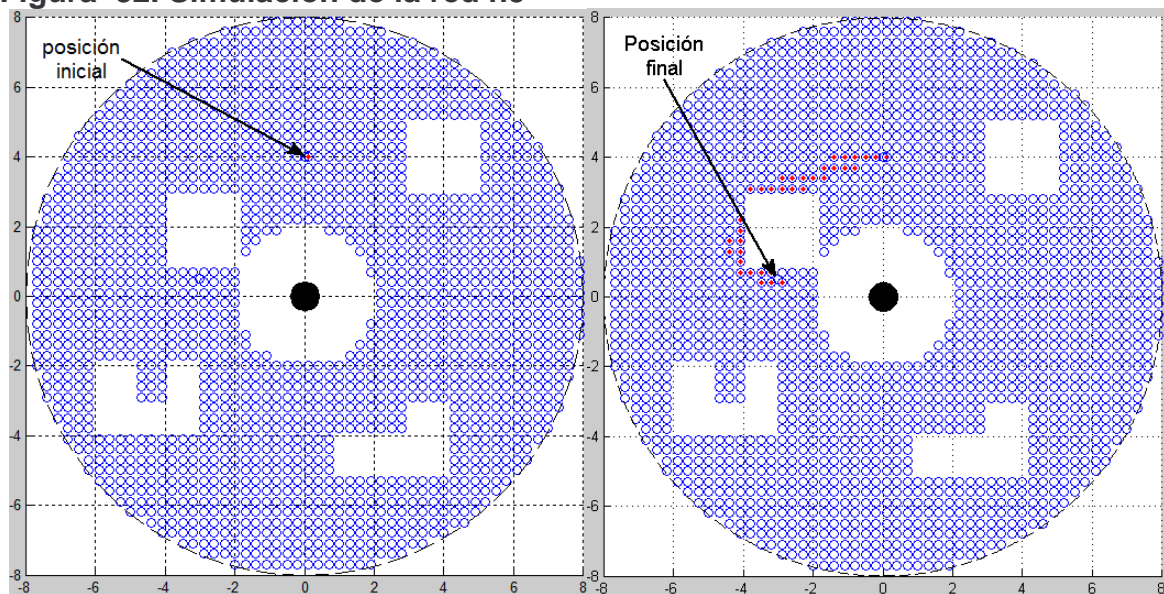
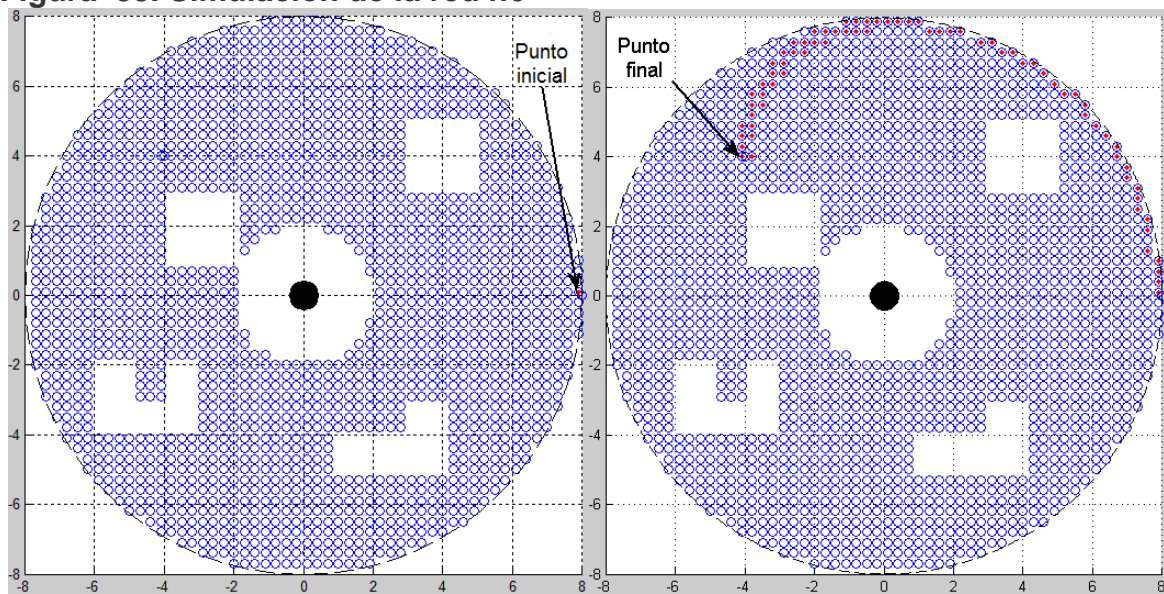


Figura 33. Simulación de la red n6



7. RESULTADOS Y CONCLUSIONES

Utilizando una estrategia de control basada en redes neuronales se logró realizar el movimiento de un brazo robot tipo Scara desde un punto inicial hasta un punto final predeterminado.

Por medio de una red neuronal se consiguió realizar un entrenamiento utilizando el algoritmo de LM para la generación de los ángulos de las articulaciones de acuerdo a una posición XY del espacio.

Utilizando la regla de aprendizaje de Kohonen se logró construir una zona de trabajo compuesto por una representación de neuronas. Dichas neuronas son distribuidas a lo largo de una región en forma de anillo, cada neurona representa un espacio en la zona.

En la construcción de la zona de trabajo se pudo establecer obstáculos que son representados por neuronas inhibidas que no presentan movimiento permaneciendo en la posición inicial.

Se consiguió que la red neuronal utilizando el criterio de vecindad, construyera una trayectoria óptima en una zona de trabajo presente de obstáculos.

Por medio de un diseño mecánico del brazo se consiguió un buen factor de seguridad utilizando el criterio de falla por Tresca, realizando un análisis en el software de Algor V22.

BIBLIOGRAFÍA

Bickford, William B. Mecánica de sólidos conceptos y aplicaciones. Madrid: McGraw-Hill, 1995. 186 p.

Daza P Sandra Patricia. Redes neuronales artificiales Fundamentos, modelos y aplicaciones [en línea]. Buenos Aires: Monografía S.A, 1995 [consultado 5 de Noviembre de 2008]. Disponible en Internet:
<http://www.monografias.com/trabajos12/redneur/redneur.shtml>

Goldberg R, Dumitru, Tailor R, Stoinanovic. A modular robotic system for ultrasound image acquisition [en línea]. Berlin: Proceeding of the 4th International Conference on Medical image Computing and Computer-Assisted intervention, 2001 [consultado 8 Octubre de 2008]. Disponible en Internet:
<http://www.springerlink.com/content/ppxl2aueqc5dq9t8/>

GOMEZ Eduardo, CAMPILLO Javier, GUTIERREZ Abel. Laboratorio de redes neuronales artificiales (RNALAB) [en línea]. Cartagena de Indias: Universidad Tecnológica de Bolívar, 2004 [consultado 9 de Octubre de 2008]. Disponible en Internet:
http://ingenieria.unitecnologica.edu.co/files/u1/ED1_4.pdf

Gutiérrez C, Ricardo. Aplicación de inteligencia artificial en el control de manipuladores teleoperados [en línea]. Bogotá D.C: Universidad Nacional de Colombia [consultado 13 de Octubre de 2008]. Disponible en Internet:
http://ricardomsc.galeon.com/pdf/est_arte_b.pdf

HACE Ales, SAFARIE Riko, KAREL Jezernik. Artificial neuronal networks control for manipulators and lyapunov theory [en línea]. Maribor: University of Maribor, 1995 [consultado 9 de Octubre de 2008]. Disponible en Internet:
<http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.8.1293>

MENDOZA José. Diseño del control de un robot de dos grados de libertad para aplicaciones de seguimientos de objetos [en línea]. Santa María Tonantzintla: Instituto Nacional de Astrofísica. Facultad de Óptica y Electrónica, 2003 [consultado 8 de Octubre de 2008]. Disponible en Internet:
<http://delta.cs.cinvestav.mx/~matias/PDF/Protocolo.pdf>

MORAL Analía, SORIA Carlos, MUT Vicente. Control de un manipulador tipo scara aplicado a tareas en interacción con sistemas biológicos [en línea]. San Juan: Instituto de Automática Universidad Nacional de San Juan, 2006 [consultado 5 de Noviembre de 2008]. Disponible en Internet:
<http://www.frc.utn.edu.ar/jar2006/docs/Papers/016-jar06.pdf>

Ricardo E, Gutiérrez C. Control de Manipuladores Teleoperados [en línea]. Bogota: Universidad Nacional de Colombia, 2003 [consultado 8 de Octubre de 2008]. Disponible en Internet:
<http://www.umng.edu.co/www/resources/rev16.1%20art03.pdf>

Vazquez Fernando, Rafael Eduardo, Jaramillo Tania, Ocotitla Alberto. Brazo manipulador tipo scara para clasificación de piezas utilizando información visual [en línea]. Mexico: Unidad Profesional Interdisciplinaria de Ingeniería y Ciencias Sociales y Administrativas, 2007 [consultado 1 de Octubre de 2008]. Disponible en Internet:
http://ocobot.googlepages.com/brazo_SCARA_UPIICSA.pdf

VIVAS A Oscar A. Control predictivo de un robot tipo scara. En: Revista Chilena de Ingeniería. Agosto, 2006, vol. 14, no. 2, p. 135 – 145.